# CHAPTER 21

# GRAPH THEORY AND NETWORK MODELS

**Dr.V.Maheswari**

**Professor**

**Department of Mathematics**

**Vels Institute of Science, Technology and Advanced Studies,**

**Pallavaram, Chennai, Tamil Nadu.**

**Email: mahaprabu76@gmail.com**

## Abstract

Graph theory is a fundamental area of discrete mathematics that studies structures formed by vertices and edges, providing powerful tools for modeling pairwise relationships. Network models, built upon graph-theoretic principles, are widely used to represent and analyze real-world systems such as communication networks, transportation systems, social networks, biological interactions, and information flow. This chapter introduces the foundations of graph theory, key concepts and types of graphs, classical graph algorithms, and major network models. Emphasis is placed on the practical relevance of graph theory in modeling, analyzing, and optimizing complex networks across disciplines.

## Keywords

Graph Theory; Network Models; Graph Algorithms; Connectivity; Shortest Path; Flow Networks; Applications of Graphs

## 1. Introduction

Modern scientific and technological problems often involve complex systems composed of interacting components. **Graph theory** provides a mathematical framework for representing such systems in terms of nodes (vertices) and connections (edges). Originating from Euler's solution to the Königsberg bridge problem, graph theory has grown into a central area of mathematics with

extensive applications in science, engineering, and social sciences (Bondy & Murty, 2008).

**Network models**, based on graph theory, enable the analysis of structure, efficiency, robustness, and dynamics of interconnected systems.

## 2. Basic Concepts of Graph Theory

### 2.1 Definition of a Graph

A graph GGG is an ordered pair G=(V,E)G = (V, E)G=(V,E), where:

- VVV is a non-empty set of vertices (nodes)
- EEE is a set of edges connecting pairs of vertices

Edges may be **undirected** or **directed**, leading to different types of graphs (West, 2001).

### 2.2 Types of Graphs

Common types of graphs include:

- **Simple graphs**
- **Multigraphs**
- **Directed graphs (digraphs)**
- **Weighted graphs**
- **Complete graphs**
- **Bipartite graphs**

Each type is suited to modeling specific real-world problems.

## 3. Fundamental Graph Properties

### 3.1 Degree of a Vertex

The degree of a vertex is the number of edges incident to it. In directed graphs, **in-degree** and **out-degree** are considered.

### 3.2 Paths, Cycles, and Connectivity

- A **path** is a sequence of vertices connected by edges.
- A **cycle** is a closed path with no repeated vertices (except start and end).
- A graph is **connected** if there exists a path between every pair of vertices.

Connectivity plays a crucial role in network reliability and communication.

### 3.3 Subgraphs and Isomorphism

A **subgraph** is formed from a subset of vertices and edges of a graph. Two graphs are **isomorphic** if they have the same structure, even if their vertex labels differ.

## 4. Special Graph Classes

### 4.1 Trees

A **tree** is a connected graph with no cycles. Trees are widely used in data structures, hierarchical modeling, and network design.

Key properties:

- A tree with nnn vertices has n−1n - 1n−1 edges
- There is a unique path between any two vertices

### 4.2 Planar Graphs

A graph is **planar** if it can be drawn in a plane without edge crossings. Planar graphs are important in circuit design and geographical mapping.

## 5. Graph Algorithms

Graph algorithms provide computational tools for solving network problems.

### 5.1 Shortest Path Algorithms

- **Dijkstra's algorithm** – shortest path in weighted graphs
- **Bellman–Ford algorithm** – handles negative edge weights

Used in routing and navigation systems (Cormen et al., 2022).

### 5.2 Minimum Spanning Tree

Algorithms such as:

- **Prim's algorithm**
- **Kruskal's algorithm**

are used to design cost-efficient networks such as electrical grids and communication networks.

### 5.3 Graph Traversal

- **Breadth-First Search (BFS)**
- **Depth-First Search (DFS)**